

## **ALIGNMENT OF A FLIGHT VEHICLE BASED ON RECURSIVE MATRIX INVERSION**

### **Background Information**

[0001] Weapons systems have been developed that include reentry bodies that with guidance and navigation systems to control reentry of the body after separation from a launch vehicle. To accurately control the trajectory of the reentry body, the guidance and navigation system of the reentry body uses position, velocity and orientation information. Typically, the reentry body has an inertial measurement unit (IMU) to provide data to the guidance and navigation system during reentry.

[0002] The launch vehicle typically also has a guidance and navigation system along with appropriate sensors, e.g., an IMU, etc. The IMU in the launch vehicle has its own reference frame; commonly called the inertial frame (I-frame). Similarly, the IMU in the reentry vehicle has its own reference frame; commonly referred to as the pseudo-inertial frame (P-frame). To allow the reentry vehicle to properly navigate after separation from the launch vehicle, flight systems commonly determine the orientation of the P-frame with respect to the I-frame using, e.g., a Kalman filter. The Kalman filter outputs information used by the reentry body's navigation computer to determine the vehicle position and velocity in the reference I-frame.

[0003] Unfortunately, the Kalman filter is complex and difficult to implement. Specifically, the Kalman filter requires an initial estimate of the relative orientations of the I-frame and P-frame such that the small angle approximation is valid. A poor initial estimate could lead to divergence of the Kalman filter even when there is complete observability of the relative orientations.

[0004] Therefore, there is a need in the art for an alignment mechanism in a flight vehicle that does not require an a priori estimate that is good enough for the small angle approximation to be valid.

### **Summary**

[0005] Embodiments of the present invention provide an improved alignment mechanism for flight vehicles. The alignment mechanism is based on a recursive matrix inversion algorithm. In one embodiment, the algorithm uses a weighting function to improve performance. In one embodiment, the weighting function is based on the magnitude of the cross product between the

pseudo position and velocity vector and, in another embodiment, the weighting function includes the angular separation between these vectors.

[0006] In one embodiment, a method for recursively determining alignment of a flight vehicle during flight is provided. The method includes generating data in a reference coordinate frame and in a second coordinate frame at a plurality of points in time during the flight, recursively generating first and second matrices from the data in the reference coordinate frame and the second coordinate frame, and at each point in time, determining an alignment output based on the inverted first matrix and the second matrix.

#### Brief Description of the Drawings

[0007] Figure 1 is a data flow diagram of one embodiment of a process for determining alignment of a flight vehicle based on recursive matrix inversion.

[0008] Figure 2 is a block diagram of one embodiment of a system that determines alignment of a flight vehicle based on recursive matrix inversion.

#### Detailed Description

[0009] In the following detailed description, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration specific illustrative embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical and electrical changes may be made without departing from the spirit and scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense.

#### **1. Introduction**

[0010] Embodiments of the present invention use a recursive matrix inversion based algorithm to align, for example, the P-frame of a reentry body with the I-frame of a launch vehicle. Advantageously, the recursive matrix inversion based algorithm does not assume a small angle approximation. Further, it also does not require an a priori estimate that is good enough for the small angle approximation to be valid. Hence, the performance of the algorithm is determined by the observability rather than the quality of the a priori estimate. This is an

improvement over existing systems using a Kalman filter since, with the Kalman filter, a poor initial estimate could lead to divergence of the filter even when there is complete observability.

## 2. Generalized Formulation

[0011] To derive the alignment algorithm, let

$$\underline{u}_i^I = \left[ \underline{a}_i^I, \underline{b}_i^I, \underline{a}_i^I \times \underline{b}_i^I \right] \quad (1)$$

[0012] and

$$\underline{u}_i^P = \left[ \underline{a}_i^P, \underline{b}_i^P, \underline{a}_i^P \times \underline{b}_i^P \right] \quad (2)$$

[0013] where  $\underline{a}$  and  $\underline{b}$  are orthogonal unit vectors.

[0014] Further define the block matrixes,

$$U_N^I = \begin{bmatrix} \underline{u}_1^I & \underline{u}_2^I & \cdots & \underline{u}_N^I \end{bmatrix} \quad (3)$$

[0015] and

$$\begin{aligned} U_N^P &= \begin{bmatrix} \underline{u}_1^P & \underline{u}_2^P & \cdots & \underline{u}_N^P \end{bmatrix} \\ &= T_I^P \begin{bmatrix} \underline{u}_1^I & \underline{u}_2^I & \cdots & \underline{u}_N^I \end{bmatrix} = T_I^P U_N^I \end{aligned} \quad (4)$$

[0016] Notice that  $U_N^I$  and  $U_N^P$  are 3X3N matrixes.  $T_I^P$  is a matrix that defines the transformation from the I-frame to the P-frame. In one embodiment, the  $T_I^P$  matrix is a direction cosine matrix defining the orientation of the P-frame with respect to the I-frame. Here the P-frame is the unknown to be solved. The I-frame is the known reference frame. The  $T_I^P$  matrix is

used in the navigation computer of a reentry body to determine the vehicle position and velocity in the I-frame. Essentially, this  $T_i^P$  matrix produces the same output as a Kalman filter.

[0017] Construct the 3X3 matrixes,

$$W_{PP}(N) = U_N^P (U_N^P)^T \quad (5)$$

[0018] and

$$W_{IP}(N) = U_N^I (U_N^P)^T \quad (6)$$

[0019] The I-to-P transformation matrix can be obtained by matrix inversion,

$$T_i^P(N) = W_{PP}(N) W_{IP}^{-1}(N) \quad (7)$$

### 3. Recursive Formulation

[0020] The above equations will be expressed in a recursive form to facilitate a software implementation. It is understood, however, that the mechanism described herein can also be implemented in hardware, firmware or any appropriate combination of hardware, software or firmware.

[0021] By definition,

$$U_{N+1}^I = [u_1' \quad u_2' \quad \dots \quad u_N' \quad u_{N+1}'] = [U_N^I \quad u_{N+1}'] \quad (8)$$

[0022] Hence,

$$W_{PP}(N+1) = U_{N+1}^P (U_{N+1}^P)^T = [U_N^P \quad u_{N+1}^P] [U_N^P \quad u_{N+1}^P]^T = W_{PP}(N) + u_{N+1}^P (u_{N+1}^P)^T \quad (9)$$

[0023] Similarly,

$$W_{IP}(N+1) = W_{IP}(N) + u_{N+1}^I (u_{N+1}^P)^T \quad (10)$$

[0024] The I-to-P transformation matrix that represents the P-frame alignment with respect to the I-frame, can be solved through inverting a 3X3 matrix as follows,

$$T_I^P(N+1) = W_{PP}(N+1) W_{IP}^{-1}(N+1) \quad (11)$$

#### 4. Choice of a and b Vectors

[0025] The alignment algorithm developed above involves the computation of the orthogonal unit vectors  $\mathbf{a}$  and  $\mathbf{b}$  using IMU measurements of the reentry body in the P-frame and measurements from a reference system in the I-frame. The orthogonal unit vectors could be the sensed acceleration and angular rate with the reference being the nominal trajectory. For better accuracy, the reference system could be the based on outputs of the IMU in the launch vehicle. Typically, the IMU in the launch vehicle is a stabilized platform mechanization. In one embodiment, angular rate is derived from gimbal resolver data. For concept evaluation, an algorithm evaluation test bed was developed using existing flight data from prior missile tests.

#### 5. Weighting Functions

[0026] The performance characteristics presented in Figure 1 and the trajectory characteristics presented in Figures 2 and 3, suggest not all the data are not equally useful in providing the alignment information. Thus, in one embodiment, the alignment algorithm uses weighting on the data. Equations (9), (10), and (11) derived above become,

$$W_{PP}(N+1) = U_{N+1}^P (U_{N+1}^P)^T = \begin{bmatrix} U_N^P & w_{N+1} u_{N+1}^P \end{bmatrix} \begin{bmatrix} U_N^P & w_{N+1} u_{N+1}^P \end{bmatrix}^T = W_{PP}(N) + w_{N+1}^2 u_{N+1}^P (u_{N+1}^P)^T \quad (13)$$

where  $w_{N+1}^2$  is the weighting function assigned to the data available at time  $t_{N+1}$

[0027] Similarly,

$$W_{IP}(N+1) = W_{IP}(N) + w_{N+1}^2 u_{N+1}^I \left( u_{N+1}^P \right)^T \quad (14)$$

[0028] The I-to-P transformation matrix that represents the P-frame alignment with respect to the I-frame, remain the same as follows,

$$T_I^P(N+1) = W_{PP}(N+1) W_{IP}^{-1}(N+1) \quad (15)$$

## 6. Implementation System

[0029] Figure 1 is a data flow diagram of one embodiment of a process indicated generally at 100 for determining alignment of a flight vehicle based on recursive matrix inversion. The process generates data in an I-frame from a reference IMU at block 102. Typically, the reference IMU is located on a launch vehicle. The data is typically pseudo position and velocity data although other data could also be used as described above. At block 104, similar data is generated in the P-frame with a slave IMU, e.g., an IMU located on a reentry body. In one embodiment, the data at block 102 and 104 is generated every 1 second during at least a portion of the flight. In other embodiments, other appropriate time intervals are used.

[0030] At blocks 106 and 108, matrices are constructed in the I and P frames, respectively, based on the data generated in blocks 102 and 104. These matrices are 3x3 matrices as defined in equations (3) and (4) above.

[0031] At blocks 110 and 112, the process recursively accumulates matrices  $W_{IP}$  and  $W_{PP}$ . In one embodiment, the  $W_{IP}$  matrix is developed recursively according to one of equations (10) and (30) above. Further, the  $W_{PP}$  matrix is similarly developed according to one of equations (9) and (29) above. In one embodiment, the weighting function used with equations (30) and (29) is the sine of the angular separation between the pseudo position and velocity. In other embodiments, the weighting function is based on the magnitude of the angular separation between the pseudo position and velocity vectors.

[0032] At block 114, the process inverts the  $W_{IP}$  matrix is inverted. Further, at block 116, the I frame to P frame transformation matrix is calculated based on one of equations (11) and (15). In one embodiment, this matrix represents a direction cosine matrix which defines the

orientation of the P-frame with respect to the I-frame. This matrix is used in a navigation computer to determine the position and velocity of a reentry body in the reference I-frame.

[0033] Figure 2 is a block diagram of one embodiment of a flight system, indicated generally at 200, that determines alignment of a flight vehicle based on recursive matrix inversion. In one embodiment, the flight system 200 includes a first flight vehicle 201, e.g., a launch vehicle, and a second flight vehicle 203, e.g., a reentry body. First flight vehicle 201 includes a master inertial measurement unit (IMU) 202 that generates data in a reference frame commonly referred to as the I-frame. The second flight vehicle 203 includes slave IMU 204 that generates data in a second coordinate frame referred to as the P-frame.

[0034] The second flight vehicle 203 processes data from the reference IMU 202 and the slave IMU 204 at a selected interval, e.g., every second, to determine the relationship between the I and P frames. This processing is performed in alignment processor using recursive matrix inversion 206. In one embodiment, the alignment processor 206 implements equations (9) – (11) above to determine the relationship between the I and P frames. In other embodiments, the alignment processor uses the relationship in equations (13) – (15) to determine the relationship between the I and P frames. The output of the alignment processor 206 is fed to the navigation computer 208 of the second flight vehicle 203, e.g., a reentry body, for use in controlling the trajectory of the reentry body.

[0035] The methods and techniques described here may be implemented in digital electronic circuitry, or with a programmable processor (for example, a special-purpose processor or a general-purpose processor such as a computer) firmware, software, or in combinations of them. Apparatus embodying these techniques may include appropriate input and output devices, a programmable processor, and a storage medium tangibly embodying program instructions for execution by the programmable processor. A process embodying these techniques may be performed by a programmable processor executing a program of instructions to perform desired functions by operating on input data and generating appropriate output. The techniques may advantageously be implemented in one or more programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system or “machine readable medium,” at least one input device, and at least one output device. Generally, a processor will receive instructions and data from a machine readable medium such as a read-only memory

and/or a random access memory. Storage devices or machine readable medium suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and DVD disks. Any of the foregoing may be supplemented by, or incorporated in, specially-designed application-specific integrated circuits (ASICs).

[0036] A number of embodiments of the invention defined by the following claims have been described. Nevertheless, it will be understood that various modifications to the described embodiments may be made without departing from the spirit and scope of the claimed invention. Accordingly, other embodiments are within the scope of the following claims.